23. (Amended) In a computing system in which a group[s] of individual instructions [are] is executable in parallel by processing pipelines, a method for supplying each individual instruction in a group to be executed in parallel to an appropriate processing pipeline, the method comprising:

storing in storage an instruction frame, the frame including at least one group of individual instructions to be executed in parallel, each individual instruction in the group having associated therewith a pipeline identifier indicative of the processing pipeline which will execute that individual instruction and a group identifier indicative of the group identification;

comparing the group identifier of each individual instruction in the instruction frame [and a] with an execution group identifier of those instructions to be next executed in parallel; and

using the pipeline identifier of those instructions to be next executed in parallel to control switches in a crossbar switch having a first set of connectors coupled to the storage for receiving instructions therefrom and a second set of connectors coupled to the processing pipelines to thereby supply each individual instruction in the group to be executed in parallel to the appropriate processing pipeline.

REMARKS

The Examiner's Action has been received and reviewed by counsel for Assignee. In that Action all claims were rejected under 35 U.S.C. §112, and claims 1-9, 20 and 21 were rejected under 35 U.S.C. §102(e). Claims 10-19, 22 and 23 were indicated as being allowable if certain Section 112 rejections were overcome.

The Examiner's careful attention to the application and claims is appreciated. By this response, extensive amendments have been made to almost all of the claims. As a result, all claims are now believed to be placed in condition for allowance.

A discussion of each of the Examiner's rejections in the manner in which it is overcome appears below.

Drawings

The drawings submitted were informal drawings. Now that allowable subject matter has been indicated, counsel proposes to submit formal drawings before mid-November, 1994. These formal drawings will correct any typographical, inconsistent labels, difficult-to-read notations, etc.

Section 112 Rejections

In the following discussion the paragraphs are numbered to correspond to the numbering of the paragraphs of the Office Action.

3a. The Examiner has inquired about a phrase used in the preambles of many of the claims relating to the parallel execution of groups of instructions. The primary goal of Applicants' invention is to permit the execution of a group of instructions in parallel, which instructions are maintained in a register, cache memory, or other apparatus, without regard to the presence of additional instructions therein. Figure 10 shows a typical register including eight instruction words divided into three groups. In a preferred embodiment of the invention, all of the instructions belonging to Group 0 will be executed first, then all of the instructions belonging to Group 1, and finally all of the instructions belonging to Group 2. As described in the specification, the size of the groups is arbitrary, and typically will range from one instruction to the number of

instructions which will fit within the register, cache line, etc. Thus, the goal of the invention is to execute each individual instruction in a group in parallel, and then continue with the next group. A suitable clarification has been made to the preambles of the appropriate independent claims (claim 1, 10, 16, 20 and 23.)

3b. The Examiner rejected claim 1 as an example of a claim in which not all elements had been positively recited, e.g., the pipeline. The claims have been carefully reviewed and numerous amendments made which are believed to overcome this rejection. In particular, a positive recitation of a processing pipeline has been made in the preamble of claim 1.

3c. The Examiner rejected two of the means clauses of claim 1 on the basis that they did not recite sufficient structure to accomplish the function. By this amendment counsel has added additional terminology to these means clauses, as well as to means clauses elsewhere in the claims, which is believed sufficient to overcome this rejection.

3d. The Examiner has rejected certain claims as being inconsistent with each other or incomplete in their terminology. By this response counsel has corrected some of those inconsistencies; however, many were intended. For example, it is desired that claim 1 be in means-plus-function format, and that claim 11 not be in means-plus-function format. Accordingly, the phrase "storage means" was used in claim 1, while "storage" was used in claim 11. The use of the different terminology in the different claims is believed permissible, and is used here because of the continuing evolution of the interpretation of means-plus-function claims under the most recent CAFC decisions.

The Examiner's comment with respect to crossbar is appreciated, and in claim 11 "switch" has been added.

The Examiner's observation about the uses of "pipelines" and "processing pipelines" as interchangeable is correct. Counsel has attempted to uniformly characterize the pipelines throughout the claims as "processing pipelines." Additionally, the preamble of claim 11 specifically rejected on this basis has been amended to directly call for "a set of processing pipelines."

4. The Examiner has indicated that claims 10-23 would be allowable if rewritten to overcome the Section 112 rejections. By this amendment those claims have been revised in a manner which is believed to overcome these rejections. The particular rejections are described below.

4a. The Examiner has inquired about what constitutes group and pipeline identifiers and how they are associated with particular instructions. The group and pipeline identifiers are described throughout the specification, for example, beginning on page 8 at line 15. In summary, instructions which can be dispatched to the parallel pipeline simultaneously are known as groups, and are identified as such at the time the instructions are compiled, and then tagged with a group identification tag. As also described in the specification, this tag may consist of a field appended to the instruction itself, or may consist of a field in some other register which is tracked and associated with a particular instruction for later use when that instruction is to be executed.

Instructions are also tagged with a pipeline tag. This tag is indicative of the specific processing pipeline to which that instruction will be dispatched, when its group is dispatched, when its group is dispatched. The pipeline tag associated with a given instruction is also typically appended to the instruction; however, the pipeline tag may also be tracked separately and then later associated with a particular instruction at execution time. Because the group identifiers and pipeline identifiers need not physically

reside with the instruction word, the phrase "associated therewith" has been used throughout claim 10 to designate the relationships between instructions in a group, and the group identifier for each instruction in that group. A similar wording has been employed for a description for the pipeline identifiers.

In addition, the Examiner noted the presence of the phrase "execution unit" in claim 10, and this has been deleted.

4b. The Examiner raised a question with respect to claim 11 concerning the phrase "receiving instructions." For clarity the crossbar switch has been characterized as "transferring instructions."

4c. The Examiner has inquired about claims 12 and 17, and the number of communication buses corresponding to the number of instructions. These claims are intended to cover the storage for the set of instructions awaiting execution. In other words, the storage corresponding to register 130 in Figure 10. As described in the specification, the storage can consist of a line in the instruction cache, a register, or some other memory which contains, in the preferred embodiment, the next eight instructions awaiting execution. When storage is interpreted in this manner, as shown by Figure 11, there are indeed corresponding numbers of communication buses.

The Examiner has also noted ambiguity with respect to the manner in which the number of decoders is claimed, as well as the switch signals. Appropriate amendments have been made to claims 12 and 17 to clarify these claims. The Examiner's careful attention to the claims is appreciated.

4d. The Examiner has inquired about the specific interconnections of the detector means in claim 13. Because counsel wishes to have the claim cover both the circumstance in which the detector means is coupled to the storage, and receives from the storage the group identifier, and also

cover other embodiments in which the detector means receives the group identifier, for example, directly from the compiler or somewhere else, no amendment has been made to the claim to specify the coupling of the detection means. Counsel believes it is sufficient to require that the detection means be coupled to receive the group identifier.

4e. The Examiner has inquired about the specific language regarding the multiplexer in claims 14, 15, 18 and 19. Appropriate amendments have been made to these claims to specify the operation of the multiplexer more clearly.

4f. The Examiner has inquired about the specific interconnections of the selection means and decoder means. For the same reasons as described above, counsel wishes to allow the selection means to be connected only to receive the group identification, and not specify where that group identification originates. Alternate embodiments of the invention are described in the specification in which the selection means receives a group identification from different locations. Claim 16 has been amended, however, to specify that the decoder be coupled to the selection means.

4g. Claims 20 and 21 were rejected as unclear with respect to what constitutes a pipeline identifier. In the preferred embodiment, the pipeline identifier accompanies an instruction to be executed and specifies the pipeline for execution of that instruction. The pipeline identifier, which is determined at the time of compilation in the preferred embodiment, either accompanies the instruction as an additional field, or is used at an appropriate stage to create the switch control signal to switch the instruction into the appropriate pipeline. The language of claim 20 has been clarified to specify that the pipeline identifiers for each instruction in the group be unique.

The Examiner has also rejected claim 21 as being unclear with respect to the number of decoders. The claim has been amended to clarify this issue.

4h. The Examiner has rejected claim 22 as unclear with respect to how the group identifier is produced. The production of the group identifier is not within the scope of claim 22. As described in the specification, for example, on page 4, at lines 21-32, the invention relies upon a compiler to assign instruction sequence codes which are used to sort the instructions into appropriate groups and execute them in the desired order. Because the compiler assigns the group identifier, that feature of the method has been incorporated into the preamble of claim 22 as an already existing aspect.

Section 102 Rejections

The Examiner rejected claims 1-9 under 35 U.S.C. §102(e) as being anticipated by Iizuka in U.S. Patent 5,299,321. The *Iizuka* patent appears quite different from the system invented by Applicants. Below the invention is briefly discussed, and then the Iizuka patent.

According to the invention, at the time of compilation, instructions are sorted into groups. All of the instructions within a single group can be executed in parallel. Also at the time of compilation, a pipeline identifier is assigned to each instruction descriptive of the processing pipeline to which that instruction should be assigned for execution. If individual processing pipelines optimized for load instructions, store instructions, floating point instructions, etc., are provided, then a tag can be added to the instruction designating to which of the pipelines that instruction should be sent for execution.

To overcome some of the disadvantages of many of the prior art systems, however, a series of instructions, even instructions from different groups, can be loaded into

the instruction register or maintained on the same line in the cache. Using the system described in the specification, instructions commonly assigned to a group may then be executed in parallel with appropriate processing pipelines, and then the next group of instructions can be executed. The system allows the flexibility of individual instructions being executable. For example, if eight sequential instructions are presented which cannot be executed in parallel, then those eight instructions may still be loaded into the instruction register and executed one at a time out of that register.

The Iizuka reference does not provide these capabilities. In particular, as best understood, Iizuka provides a parallel processor in which a series of processing pipelines have decoders associated with them, but which operate using a single standard width bus. Apparently, a looping function is performed to execute the parallel instruction. (See column 7, line 50, and column 8, line 58.)

Thus, in contrast to the present invention, Iizuka does not appear to provide any capability of placing in an instruction register sets of instructions which may be executed in parallel, separately, or in various combinations. As best understood, Iizuka appears to provide a system in which an instruction is decoded into 1-to-4 subinstructions, which are then executed.

With respect to claim 1, Iizuka does not appear to provide the claimed storage means in which N individual instructions may be placed in the storage means, and can include within a group of M individual instructions which are to be executed in parallel. Similarly, with respect to claim 6, Iizuka does not appear to provide the capability of placing a fixed number of the instructions in a register which number includes at least one group of instructions having the common group identifier as well as at least one

other instruction. For at least these reasons, claims 1 and 6, as well as claims 2-5 and 7-9, are believed to patentably distinguish Iizuka.

Claims 20 and 21 were rejected under 35 U.S.C. §102(e) as anticipated by Kumar, *et al.*, in U.S. Patent 5,197,137. Kumar, *et al.*, teaches a computer architecture for parallel execution of programs; however, it does not appear to show or suggest any of the basic ideas of Applicants' invention. Generally, Kumar, *et al.*, describes a situation in which instructions are dispatched and then supplied through a crossbar network to an execution unit. As best Kumar, *et al.*, can be understood, only instructions in the same group are dispatched, and there is no capability of executing only instructions from a single group if more than that group has been dispatched. Claim 20 requires that the set of instructions in storage, corresponding to register 130 in Figure 10, include instructions to be executed in parallel, together with at least one other instruction not to be executed in parallel. For at least this reason, the Kumar, *et al.*, is believed distinguished by the claims.

Accordingly, all of the claims are now believed allowable. If the Examiner would like to discuss the application by telephone, please call the undersigned.

Respectfully submitted,

Robert C Colwell

Robert C. Colwell
Reg. No. 27,431

RCC03287:mks

TOWNSEND and TOWNSEND KHOURIE and CREW
One Market Plaza
Steuart Street Tower
San Francisco, CA 94105

Telephone: 415/326-2400